

Analisis Perbandingan Performa Web Server Berbasis PHP dan Web Server Berbasis Python Twisted

Peter Winardi¹, Muhammad Imam Abdillah², Dedy Hermanto³, Eka Puji Widiyanto⁴

^{1,2}STMIK GI MDP; Jl. Rajawali No. 14, +62(711)376400/376360

³Program Studi Teknik Informatika, STMIK GI MDP Palembang

e-mail: ¹peterwinardi@mhs.mdp.ac.id, ²iabdillah@mhs.mdp.ac.id, ³dedy@mdp.ac.id,
⁴ekapujiw2002@mdp.ac.id

Abstrak

Dalam era teknologi saat ini pengembangan perangkat lunak harus memperhatikan efisiensi dalam hal penggunaan sumber daya pada sistem, terutama pada pengembangan aplikasi yang berbasis website. Terdapat beberapa parameter web response time yang dapat menyebabkan sebuah operasi mengalami delay, salah satunya adalah server processing time. Proses yang dilakukan oleh server dapat berupa akses database, komputasi, verifikasi, dan autentifikasi. Untuk melakukan proses-proses tersebut, perlu dilakukan pengembangan web server yang efisien agar dapat meminimalisir delay yang terjadi pada server processing. Pada pengujian performance load browser web server berbasis python lebih cepat dari pada web server berbasis PHP dalam merespon permintaan client berdasarkan Tabel 4.49 dan Tabel 4.50. Untuk melakukan request data dari lima client secara bersamaan sebanyak 300.000 baris diperlukan web server dengan memory minimal 8 GB agar data tersebut dapat ditampilkan pada web browser. Spesifikasi hardware yang digunakan oleh web server tidak mempengaruhi performa dari segi response time, CPU usage dan memory usage pada saat menerima request dalam bentuk teks.

Kata kunci : Web Server, Performa, PHP, Python Twisted

Abstract

In today's technological era, software development software should pay attention to efficiency in terms of resource usage on the system, especially on website-based application development. There are several web response time parameters that can cause an operation to delay, one of which is server processing time. Processes performed by the server can be database access, computing, verification, and authentication. To perform these processes, it is necessary to develop an efficient web server in order to minimize delays that occur in server processing. In the performance load test python-based web server browser faster than the PHP-based web server in response to client requests based on Table 4.49 and Table 4.50. To request data from five clients simultaneously as many as 300,000 lines required web server with minimum memory of 8 GB so that data can be displayed in web browser. The hardware specifications used by the web server do not affect performance in terms of response time, CPU usage and memory usage when receiving requests in text form.

Keywords : Web Server, Performance, PHP, Python Twisted

1. PENDAHULUAN

Pengembangan perangkat lunak harus memperhatikan efisiensi dalam hal penggunaan sumber daya pada sistem, terutama pada pengembangan aplikasi yang berbasis

website. Dikarenakan keterbatasan sumber daya yang ada pada aplikasi berbasis web, *programmer* harus handal dalam mengelola sumber daya yang ada[1].

Selain itu aplikasi berbasis web juga tidak bisa lepas dari koneksi data melalui jaringan. Koneksi data dibutuhkan pada saat aplikasi berbasis web akan berkomunikasi dengan *web server* pada saat melakukan sebuah operasi. Pada tahap ini sangat rawan terjadi *delay*. *Delay* pada sebuah operasi akan mengakibatkan gangguan pada *user-experience* seorang *user* yang menggunakan aplikasi tersebut.

Terdapat beberapa parameter *web response time* yang dapat menyebabkan sebuah operasi mengalami *delay*, salah satunya adalah *server processing time* yaitu waktu yang dibutuhkan oleh *server* untuk memproses *request* yang dikirimkan oleh *client*. Proses yang dilakukan oleh *server* dapat berupa akses *database*, komputasi, verifikasi, dan autentifikasi. Untuk melakukan proses-proses tersebut, perlu dilakukan pengembangan *web server* yang efisien agar dapat meminimalisir *delay* yang terjadi pada *server processing*.

Performa pada *web server* dapat berbeda berdasarkan bahasa pemrograman yang digunakan[2]. Oleh karena itu, dalam melakukan pengembangan *web server* kita perlu memilih bahasa pemrograman yang tepat. Bahasa pemrograman sendiri adalah bahasa buatan yang didesain untuk mengekspresikan komputasi yang dapat dilakukan oleh mesin, umumnya adalah computer[3]. Bahasa pemrograman dapat digunakan untuk mengatur perilaku mesin, untuk mengekspresikan algoritma dengan benar, atau hanya sebagai alat bagi manusia untuk berkomunikasi dengan mesin.

Salah satu bahasa pemrograman yang digunakan pada penelitian ini adalah *Hypertext Preprocessor* (PHP), dimana biasanya digunakan untuk *server-side scripting*[1]. PHP merupakan bahasa pemrograman yang *simple* namun *powerful* dan tepat untuk digunakan pada *web server*, selain itu penelitian ini menggunakan bahasa pemrograman *Python*. Dimana *Python* adalah bahasa pemrograman yang bersifat *open source*. Bahasa pemrograman ini dioptimalisasikan untuk *software quality*, *developer productivity*, *program portability* dan *component integration*[4].

Pada umumnya *Hypertext Preprocessor* (PHP) digunakan sebagai basis dari sebuah *web sever*, akan tetapi masih ada *web server* yang menggunakan *engine* lain, salah satunya *Python Twisted*. *Python Twisted* merupakan salah satu *engine web server* yang menggunakan bahasa pemrograman *Python*. Dimana tujuan dari penelitian ini adalah untuk menganalisa perbandingan performa atau *respon time* antara *web server* berbasis *PHP* dengan *web server* berbasis *Python Twisted* saat menangani *request* data dari *client* dengan jumlah data yang besar secara bersamaan. Hal ini menyebabkan perbedaan penggunaan sumber daya yang ada pada perangkat komputer, seperti jumlah *memory* dan waktu komputasi CPU (*Central Processing Unit*) yang dibutuhkan oleh bahasa pemrograman[3]. Pada *web server* hal tersebut dapat mempengaruhi *server response time* secara signifikan.

2. METODE PENELITIAN

Berikut ini tahapan-tahapan yang dilakukan dalam menganalisis perbandingan performa *web server* berbasis *PHP* dan *web server* berbasis *Python Twisted* adalah sebagai berikut :

2.1 Studi Leteratur

Pada tahap dilakukan pengumpulan studi literatur berdasarkan jurnal yang berhubungan dengan penelitian yang dilakukan dengan pokok bahasannya adalah analisis perbandingan performa *web server*. Setelah didapatkan beberapa studi literatur, kemudian hasil yang didapat dari studi literatur yang telah dikumpulkan dirangkum, tujuan dari pengumpulan studi literatur adalah agar membantu dalam mendapatkan referensi untuk penelitian ini.

2.2 Perancangan

Penelitian dimulai dari membangun sebuah topologi yang terdiri dari dua *server* dengan basis yang berbeda dan lima *client*. Dalam perancangan alur penelitian pada langkah pertama adalah membangun sebuah topologi, kemudian membuat dua buah *server* dengan basis yang berbeda serta melakukan konfigurasi. Langkah selanjutnya yaitu melakukan penginputan data pada *database* yang telah dibuat. *Dataset* pada penelitian ini didapat dari *sample employees database* yang merupakan data *testing* yang disediakan oleh MySQL. *Database* adalah sekumpulan data yang berhubungan dengan secara logika dan memiliki beberapa arti yang saling berpautan[5]. Pada *database employees* terdapat enam *table* yaitu *table departements*, *dept_emp*, *dept_manager*, *employees*, *salaries* dan *titles*. Setelah data diinput sebanyak jumlah yang ditentukan, kemudian membangun *interface web based* dimana *interface* tersebut digunakan untuk melakukan pengujian. Hasil yang didapat dari pengujian dilakukan analisa untuk mengetahui perbandingan performa antara kedua *server* tersebut. Untuk menjalankan aplikasi tentunya dibutuhkan prangkat keras (*hardware*). *Hardware* adalah infrastruktur bagi *software* yang dapat berupa komponen-komponen fisik yang terdapat dalam komputer dan memiliki fungsi untuk memastikan bahwa sebuah komputer dapat berjalan dengan baik.

2.3 Implementasi

Pada tahapan dilakukan implementasikan rancangan yang sudah dibuat pada tahapan sebelumnya dengan membangun topologi, konfigurasi dan implementasi ke dalam bentuk bahasa pemrograman serta tahap akhir melakukan pengujian.

2.4 Analisis Hasil

Pada tahapan ini dilakukan analisis untuk menghitung performa dari hasil pengujian dengan menggunakan parameter *respon time*, *memory usage*, *CPU usage*[2]. *Response time* merupakan waktu rata-rata yang diperlukan untuk menangani setiap permintaan dari *client* secara simultan[6]. Pengukuran dilakukan menggunakan *tools siege* untuk mengukur *response time* dan *tools htop* untuk mengukur *CPU usage* dan *memory usage*. Untuk membandingkan antara kedua performa web server dilihat dari hasil rata-rata dari pengujian.

3. HASIL DAN PEMBAHASAN

Pengujian *http load testing* dilakukan menggunakan *tool siege*. *Siege* adalah *tool* yang dapat digunakan oleh pengembang web untuk melakukan uji beban atau *http load* terhadap kode yang dibuat, apakah kode tersebut dapat menangani *request* dari *client* secara simultan. Hasil pengujian *http load* dapat dilihat pada Gambar 1.

```

root@imam:/home/jihan# siege -t10s -c5 localhost/twisted/10000.php
** SIEGE 3.0.8
** Preparing 5 concurrent users for battle.
The server is now under siege...
Lifting the server siege...      done.

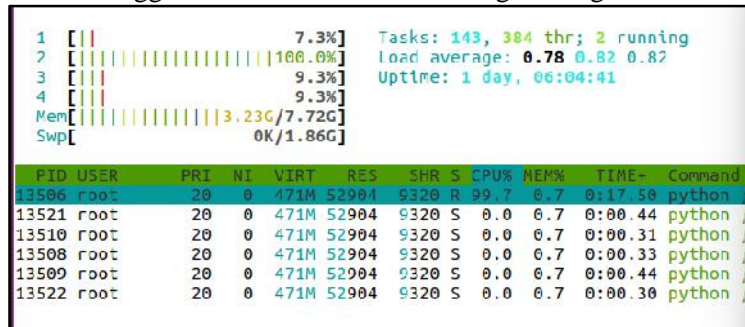
Transactions:      80 hits
Availability:      100.00 %
Elapsed time:      9.64 secs
Data transferred:  14.36 MB
Response time:     0.08 secs
Transaction rate:  8.30 trans/sec
Throughput:        1.49 MB/sec
Concurrency:       0.64
Successful transactions: 80
Failed transactions:  0
Longest transaction: 0.18
Shortest transaction: 0.05

```

Gambar 1 Hasil Pengujian *http Load*

Sedangkan untuk mengukur *CPU usage* dan *Memory usage* menggunakan *tool htop* yang dapat dilihat pada Gambar 2, dimana hasil diambil merupakan penggunaan daya tertinggi dari

nilai CPU *usage* dan *Memory usage*, dengan skala 0-400% pada nilai CPU *usage* dikarenakan server yang digunakan menggunakan 4 *core* dimana masing-masing *core* memiliki skala 0-100%.



Gambar 2 Hasil Pengujian dengan htop

Hasil pengujian didapat dari skenario yang dapat dilihat pada Tabel 1, dimana terdapat enam skenario pengujian dengan melakukan *request* data yang berbeda-beda diantaranya adalah 10000, 25000, 50000, 100000, 200000 dan 300000 baris. Setiap skenario akan diulang sebanyak lima kali.

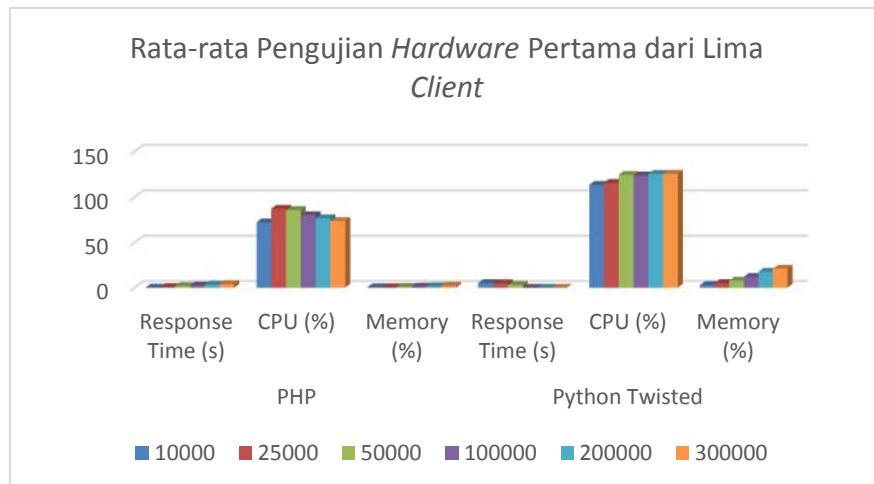
Tabel 1 Skenario Pengujian

Pengujian	Data	Sebanyak	Waktu Pengujian	Jumlah <i>Client</i>
I	10.000 baris	5 kali	10 seconds	5, 10, 20
II	25.000 baris	5 kali	10 seconds	5, 10, 20
III	50.000 baris	5 kali	10 seconds	5, 10, 20
IV	100.000 baris	5 kali	10 seconds	5, 10, 20
V	200.000 baris	5 kali	10 seconds	5, 10, 20
VI	300.000 baris	5 kali	10 seconds	5, 10, 20

Hasil dari seluruh pengujian yang dilakukan pada spesifikasi *hardware* pertama sebanyak lima kali dengan lima *client* dihitung nilai rata-ratanya pada masing-masing parameter *response time*, CPU *usage* dan *Memory usage* dapat dilihat pada Tabel 2 dan Gambar 3 untuk grafik perbandingannya.

Tabel 2 Hasil Rata-rata Pengujian dengan Spesifikasi *Hardware* Pertama dari Lima *Client*

Pengujian (baris)	PHP			Python Twisted		
	<i>Response Time</i> (s)	<i>CPU</i> (%)	<i>Memory</i> (%)	<i>Response Time</i> (s)	<i>CPU</i> (%)	<i>Memory</i> (%)
10000	0.068	103.8	0.2	1.962	37.54	1.04
25000	0.16	49.04	0.8	1.17	104.4	1.44
50000	0.466	76.92	0.4	4.704	106.8	2.48
100000	0.486	78.68	0.56	5.532	109.4	3.28
200000	0.76	72.7	0.8	0	112	4.62
300000	1.044	68.72	1.0	0	111.6	9.44

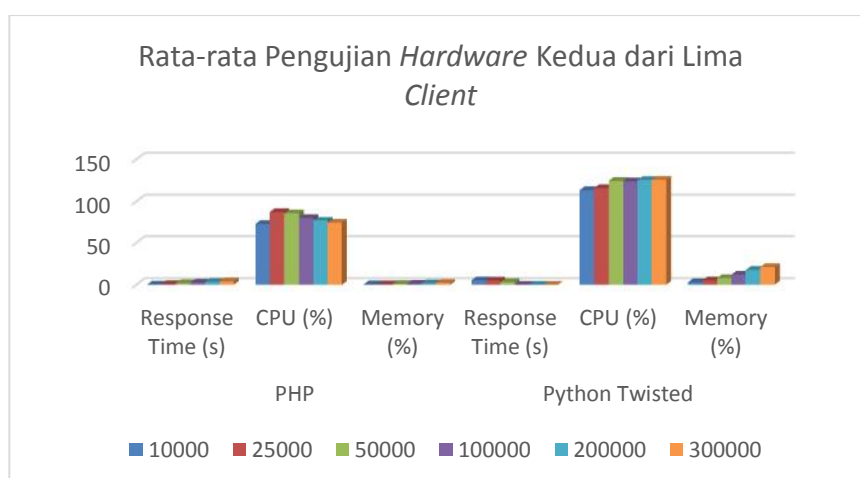


Gambar 3 Grafik Perbandingan PHP dan Python Twisted

Hasil dari seluruh pengujian yang dilakukan pada spesifikasi *hardware* kedua sebanyak lima kali dengan lima *client* dihitung nilai rata-ratanya pada masing-masing parameter *response time*, *CPU usage* dan *Memory usage* dapat dilihat pada Tabel 3 dan Gambar 4 untuk grafik perbandingannya.

Tabel 3 Hasil Rata-rata Pengujian dengan Spesifikasi *Hardware* Kedua dari Lima *Client*

Pengujian (baris)	PHP			Python Twisted		
	<i>Response Time</i> (s)	<i>CPU</i> (%)	<i>Memory</i> (%)	<i>Response Time</i> (s)	<i>CPU</i> (%)	<i>Memory</i> (%)
10000	0.068	37.88	0.58	1.928	37.88	0.58
25000	0.16	54.74	0.6	4.164	105.6	2.74
50000	0.564	88.96	0.8	5.4	108	4.64
100000	0.564	78.1	1.2	0	103.8	6.82
200000	0.762	74.98	1.64	0	113	12.84
300000	1	75.46	2.14	0	119.2	17.02

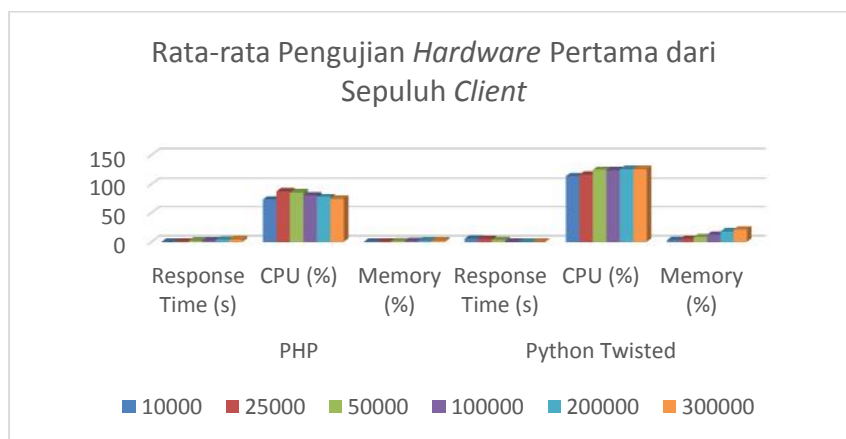


Gambar 4 Grafik Perbandingan PHP dan Python Twisted

Hasil dari seluruh pengujian yang dilakukan pada spesifikasi *hardware* pertama sebanyak lima kali dengan sepuluh *client* dihitung nilai rata-ratanya pada masing-masing parameter *response time*, *CPU usage* dan *Memory usage* dapat dilihat pada Tabel 4 dan Gambar 5 untuk grafik perbandingannya.

Tabel 4 Hasil Rata-rata Pengujian dengan Spesifikasi *Hardware* Pertama dari Sepuluh *Client*

Pengujian (baris)	<i>PHP</i>			<i>Python Twisted</i>		
	<i>Response Time</i> (s)	<i>CPU</i> (%)	<i>Memory</i> (%)	<i>Response Time</i> (s)	<i>CPU</i> (%)	<i>Memory</i> (%)
10000	0.09	55.18	0.2	4.052	110.8	1.22
25000	0.24	80.7	0.3	5.296	113.2	1.72
50000	0.79	85.84	0.4	4.5	108	2.68
100000	0.486	78.68	0.56	5.532	109.4	3.28
200000	0.76	72.7	0.8	0	112	4.62
300000	1.044	68.72	1	0	111.6	9.44



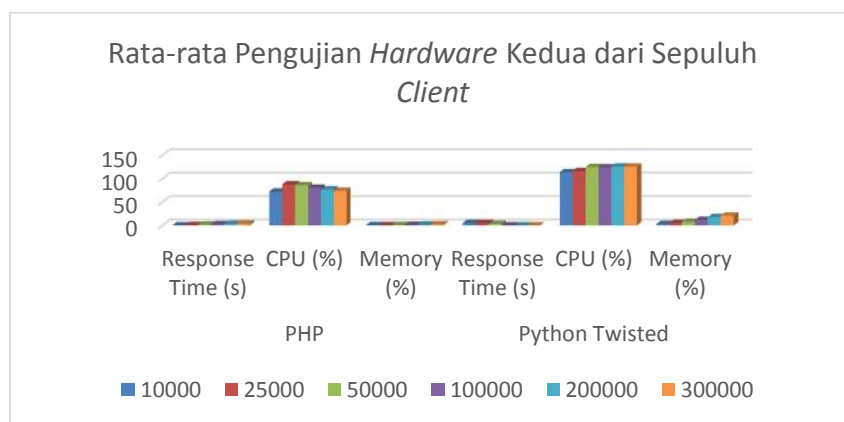
Gambar 5 Grafik Perbandingan PHP dan Python Twisted

Hasil dari seluruh pengujian yang dilakukan pada spesifikasi *hardware* kedua sebanyak lima kali dengan sepuluh *client* dihitung nilai rata-ratanya pada masing-masing parameter *response time*, *CPU usage* dan *Memory usage* dapat dilihat pada Tabel 5 dan Gambar 6 untuk grafik perbandingannya.

Tabel 5 Hasil Rata-rata Pengujian dengan Spesifikasi *Hardware* Kedua dari Sepuluh *Client*

Pengujian (baris)	<i>PHP</i>			<i>Python Twisted</i>		
	<i>Response Time</i> (s)	<i>CPU</i> (%)	<i>Memory</i> (%)	<i>Response Time</i> (s)	<i>CPU</i> (%)	<i>Memory</i> (%)
10000	0.08	54.56	0.5	4.178	111.2	8.76
25000	0.256	78.32	0.6	5.268	116.8	9.84
50000	0.772	86.9	0.8	5.56	114.2	10.9
100000	1.16	83.58	1.2	0	113.6	12.98

200000	1.774	77.14	1.62	0	122	15.64
300000	2.326	74.28	2.12	0	125	19.46

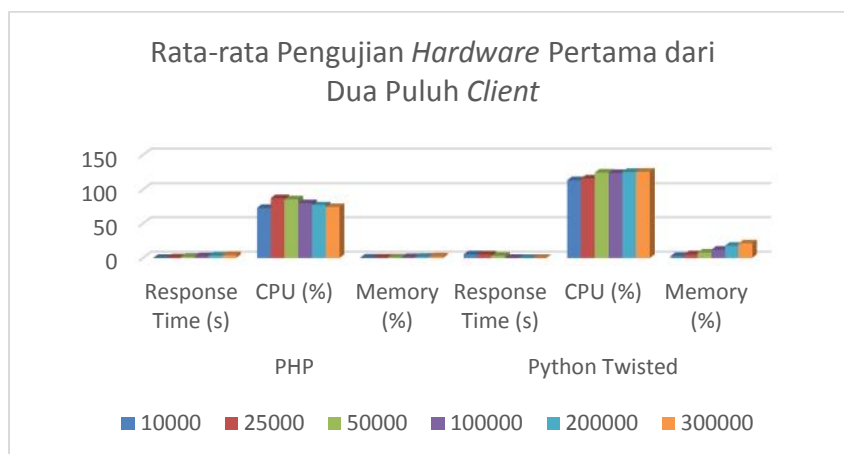


Gambar 6 Grafik Perbandingan PHP dan Python Twisted

Hasil dari seluruh pengujian yang dilakukan pada spesifikasi *hardware* pertama sebanyak lima kali dengan dua puluh *client* dihitung nilai rata-ratanya pada masing-masing parameter *response time*, *CPU usage* dan *Memory usage* dapat dilihat pada Tabel 6 dan Gambar 7 untuk grafik perbandingannya.

Tabel 6 Hasil Rata-rata Pengujian dengan Spesifikasi *Hardware* Pertama dari Dua Puluh *Client*

Pengujian (baris)	PHP			Python Twisted		
	Response Time (s)	CPU (%)	Memory (%)	Response Time (s)	CPU (%)	Memory (%)
10000	0.124	75.06	0.2	5.956	117.6	4.52
25000	0.738	86.78	0.3	3.638	155.2	5.4
50000	1.76	84.82	0.4	6.986	122	5.94
100000	2.502	83.86	0.58	0	122.4	7.52
200000	3.362	71.7	0.8	0	124.2	9.42
300000	4.076	75.06	1.06	0	124.4	10.36

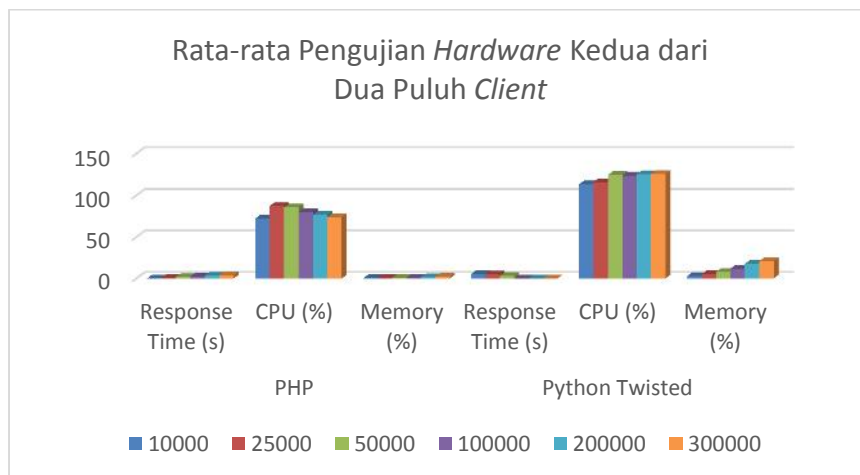


Gambar 7 Grafik Perbandingan PHP dan Python Twisted

Hasil dari seluruh pengujian yang dilakukan pada spesifikasi *hardware* kedua sebanyak lima kali dengan dua puluh *client* dihitung nilai rata-ratanya pada masing-masing parameter *response time*, *CPU usage* dan *Memory usage* dapat dilihat pada Tabel 7 dan Gambar 8 untuk grafik perbandingannya.

Tabel 7 Hasil Rata-rata Pengujian dengan Spesifikasi *Hardware* Kedua dari Dua Puluh *Client*

Pengujian (baris)	<i>PHP</i>			<i>Python Twisted</i>		
	<i>Response Time</i> (s)	<i>CPU</i> (%)	<i>Memory</i> (%)	<i>Response Time</i> (s)	<i>CPU</i> (%)	<i>Memory</i> (%)
10000	0.14	72.32	0.5	5.226	113.4	2.98
25000	0.708	87.34	0.6	5.018	115.8	5.14
50000	1.822	85.78	0.88	3.142	124.4	7.82
100000	2.53	80.14	1.14	0	123.8	11.94
200000	3.338	76.78	1.64	0	125.4	17.62
300000	3.964	73.9	2.14	0	125.6	20.96



Gambar 8 Grafik Perbandingan PHP dan Python Twisted

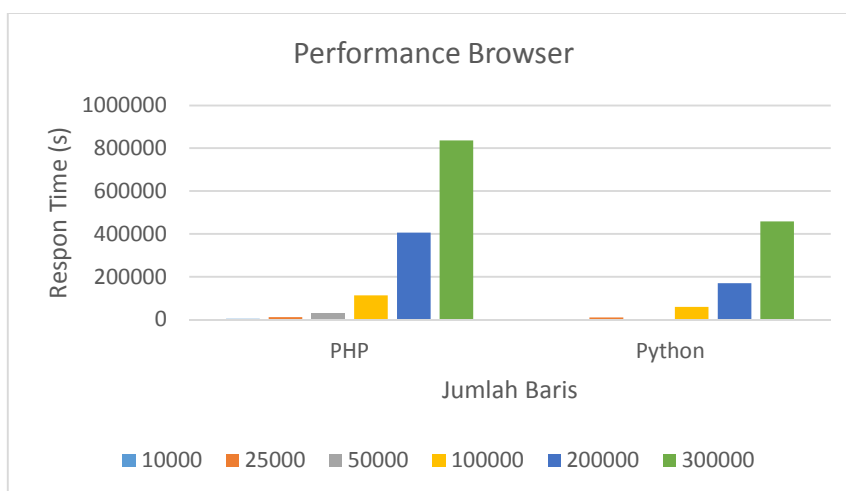
Pengujian *performance load browser* adalah pengujian yang dilakukan dengan *request* data pada web PHP dan python twisted melalui *browser*, dimana akan diukur berapa lama waktu masing-masing web dalam menangani *request* tersebut dari mulai proses *loading* sampai *loading* selesai.

Tujuan dari pengujian ini adalah untuk membandingkan hasil pengujian menggunakan *tool siege* dengan pengujian performa *browser* secara nyata. Terdapat perbedaan cara pada web PHP dan python dalam menampilkan data ke dalam *browser*, pada python data akan diakses seluruhnya terlebih dahulu setelah selesai baru akan dimuat ke dalam *browser* secara keseluruhan. Sedangkan pada PHP data akan dimuat secara bertahap sehingga membutuhkan waktu yang lebih lama dibandingkan dengan web python twisted.

Dari Tabel 8 dapat dilihat perbedaan waktu *loading* browser pada web PHP dan python twisted, dimana web python twsited membutuhkan waktu yang lebih sedikit untuk menampilkan data ke dalam *browser* dari pada web PHP. Grafik perbedaannya dapat dilihat pada Gambar 9.

Tabel 8 Hasil *Performance Browser* dari Spesifikasi *Hardware* Pertama

Pengujian (baris)	Time (ms)	
	PHP	Python
10000	4730	3368
25000	11486	10330
50000	31947	23690
100000	113916	60354
200000	406312	169657
300000	837035	458773

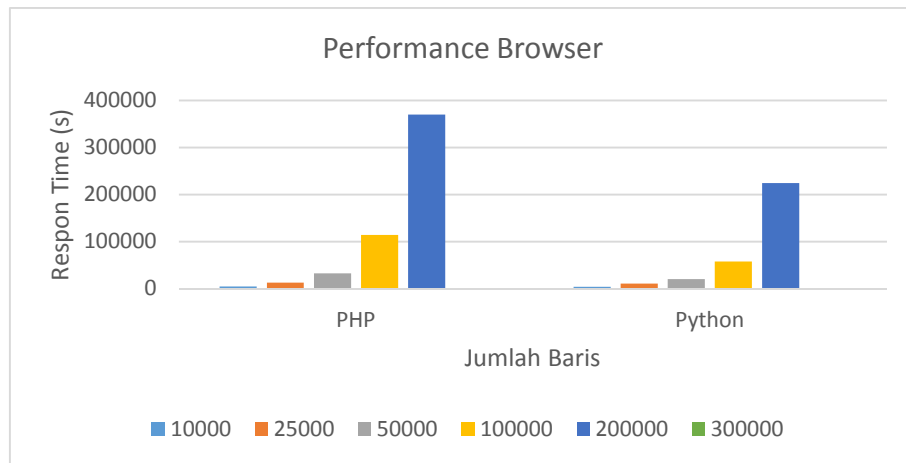


Gambar 9 Grafik Perbandingan *Performance Browser* PHP dan Python

Dari Tabel 9 dapat dilihat perbedaan waktu *loading* browser pada web PHP dan python twisted, dimana web python twsited membutuhkan waktu yang lebih sedikit untuk menampilkan data ke dalam *browser* dari pada web PHP dan tidak mengalami perubahan yang signifikan apabila dibandingkan dengan web server dengan spesifikasi *hardware* pertama, akan tetapi pada saat melakukan pengujian dengan data 300.000 server tidak merespon dengan baik sehingga *browser* menjadi *not responding*. Grafik perbandingan *performance browser* PHP dan Python dapat dilihat pada Gambar 10.

Tabel 9 Hasil *Performance Browser* dari Spesifikasi *Hardware* Kedua

Pengujian (baris)	Time (ms)	
	PHP	Python
10000	4402	3807
25000	12334	10945
50000	32048	19828
100000	114070	57205
200000	369627	224301
300000	-	-



Gambar 10 Grafik Perbandingan *Performance Browser* PHP dan Python

Dari hasil pada Gambar 10 dapat disimpulkan bahwa pada pengujian performa menggunakan *browser* web python twisted dari spesifikasi *hardware* pertama lebih cepat dibandingkan dengan web PHP. Hasil ini tidak berbeda jauh dengan pengujian menggunakan *web server* dengan spesifikasi *hardware* pertama.

4. KESIMPULAN

Setelah melakukan analisis perbandingan performa *web server* menggunakan parameter *response time*, *CPU usage* dan *memory usage* dan berdasarkan pembahasan pada bab-bab sebelumnya, maka dapat ditarik kesimpulan sebagai berikut :

1. Berdasarkan hasil rata-rata penelitian (pada Tabel 2 sampai Tabel 7), maka dapat disimpulkan bahwa *web server* berbasis *PHP* lebih efisien dari segi penggunaan *CPU* dan *memory* dikarenakan hasil analisis *response time*, *CPU usage* dan *memory usage* *web server* berbasis *PHP* dengan hasil lebih kecil dibandingkan dengan *web server* berbasis *python*.
2. Pada pengujian *performance load browser*, *web server* berbasis *python* lebih cepat dari pada *web server* berbasis *PHP* dalam merespon permintaan *client* berdasarkan Tabel 8 dan Tabel 9, tetapi *web server* berbasis *python* tidak lebih efisien dibandingkan dengan *web server* berbasis *PHP*.
3. Untuk melakukan *request* data dari lima *client* secara bersamaan sebanyak 300.000 baris diperlukan *web server* dengan *memory* minimal 8 GB agar data tersebut dapat ditampilkan pada *web browser*.
4. Spesifikasi *hardware* yang digunakan oleh *web server* tidak mempengaruhi performa dari segi *response time*, *CPU usage* dan *memory usage* pada saat menerima *request* dalam bentuk teks.

5. SARAN

Saran yang dapat diberikan untuk pengembangan lebih lanjut dari penelitian ini adalah :

1. Gunakan bahasa pemrograman lain untuk menguji performa *web server* selain *python* dan *PHP*.
2. Menggunakan *memory* diatas 8 GB apabila ingin melakukan *request* data lebih dari 200.000 baris dari lima *client* karena pada saat menampilkan data pada *web browser* menggunakan *memory* 4 GB *web browser* mengalami *not responding*.
3. Gunakan *interface* selain *web browser* untuk menampilkan hasil dari *request* data.

DAFTAR PUSTAKA

- [1] Harismawan, F, A, Kharisma, P, A, Afiriando, T., (2018). *Analisis Perbandingan Performa Web Service Menggunakan Bahasa Pemrograman Python,PHP, dan Perl pada Client Berbasis Android*. Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer.
 - [2] Ortiz, P, R, Gil, A, J, Sahuquillo, J, Pont, A., (2013). *Analyzing web server performance under dynamic userworkloads*. ELSEVIER.
 - [3] Oguntunde, Oyenike, B., (2012). *Comparative Analysis of Some Programming Languages*. *Transnational Journal of Science and Technology*, Volume 2, pp. 107-118.
 - [4] Lutz, M., (2010). *Programming Python*. Fourth Edition ed. Sebastopo: O'Reilly Media, Inc.
 - [5] Prasetyo, B, Pattiasina, J, T, Soetarmono, N, A., (2015). *Perancangan dan Pembuatan Sistem Gudang*(Studi Kasus : PT.PLN (Persero) Are Surabaya Barat). *Teknika* Volume.4 Nomor 1.
 - [6] Mahbub, C, (2009). *Benchmark Performa Web Server dengan Siege*.
-